ARMY RESEARCH LABORATORY

A Statistical Approach to the Generation of a Database for Evaluating OCR Software

by Frederick S. Brundick, Ann E. M. Brodeen,
and Malcolm S. Taylor

ARL-TR-2265 July 2000

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Abstract

In this report, we consider a statistical approach to augment a limited database of groundtruth documents for use in evaluating optical character recognition (OCR) software. We require groundtruth documents to assign a performance measure to the OCR component of the Forward Area Language Converter (FALCon) system. A modified moving-blocks bootstrap procedure is used to construct surrogate documents for this purpose which prove to serve effectively, and in some regards, indistinguishably from groundtruth. The proposed method is validated through a rigorous statistical procedure.

Table of Contents

INTENTIONALLY LEFT BLANK.

List of Figures

List of Tables

INTENTIONALLY LEFT BLANK.

1. Introduction

The Forward Area Language Converter (FALCon) is a portable, field-operated, translation system designed to assist in intelligence collection. It enables an operator with no foreign language training to convert a foreign language document into an approximate English translation for an assessment of military relevance. The principal components of FALCon are an optical scanner, an optical character recognition (OCR) module, and a machine translation (MT) module. In order to assign a performance measure to the FALCon system, measures of effectiveness of the components must be developed and then aggregated into an overall measure. The focus of this report is limited to evaluation of the OCR module.

A current procedure for determining a quantitative measure of the efficacy of an OCR product is as follows: A selection of carefully prepared source-language documents, called groundtruth, is stored in the computer; hardcopy of the same document set is then scanned into bitmap images; the OCR software partitions a gross bitmap image into homogeneous zones that are processed according to content. For zones that are identified as text, specialized scoring software then compares the OCR output against the corresponding groundtruth to produce accuracy statistics, usually including percentage agreement for both words and characters, and a confusion matrix.*

A central database of groundtruth documents, accepted as a baseline, would enable the evaluation of OCR products to proceed from a common benchmark. Unfortunately, such a database does not exist, making the comparison of OCR software more difficult and any conclusions drawn more tentative. Fundamental questions regarding sample size requirements, and suitable document composition for such a database, remain to be addressed.

Collection of a corpus that is sufficient for evaluation of an OCR product is likely to remain, even in the best of circumstances, a burdensome task. Access to a sufficient number of source-language documents, representative of the document classes of interest, may not be feasible; and, even if obtained, the expensive and time-consuming process of preparing groundtruth remains. To address this problem, we are proposing a statistical approach to corpus generation based on a small set of source-language documents. Coincident with the statistical inquiry, substantial work involving language transliteration must be accomplished.

2. Time Series Model

Consider the passage of Serbian text shown in Figure 1. Every character—letters, punctuation marks, interword spaces—is represented numerically in the computer. The set of character and numeric equivalents (the mapping) is called a codeset. For a specific language, the codeset representation may not be unique. Russian, for example, has four commonly used 8-bit encodings and some Asian languages even more (Reeder 1998). A numeric representation of the Serbian text in Figure 1 for a particular codeset is shown in Figure 2.

*A confusion matrix displays the number of character insertions, substitutions, and deletions required to reconcile the groundtruth and OCR output files.

Баш тај недостатак суштинске разлике у религиозној самосвести секташа нам и допушта да говоримо о нечем, што би иначе било недопустиво о сектама ен генерал.

Религиозна или секташка самосвест се затим изражава и кроз ренебрегавање бриге за овај свет са упорним инсистирањем да се "праве ствари" дешавају тек "тамо", истакнути апокалиптизам ишекивање краја света, његове пропасти, презир према телу и целокупној материји који се манифестује или као ригорозни и бесмислени аскетизам који постаје циљ сам по себи или као радикални разврат који су у суштини идентични феномени не прихватања тела и материје уопште и сл.

Figure 1. Serbian text.

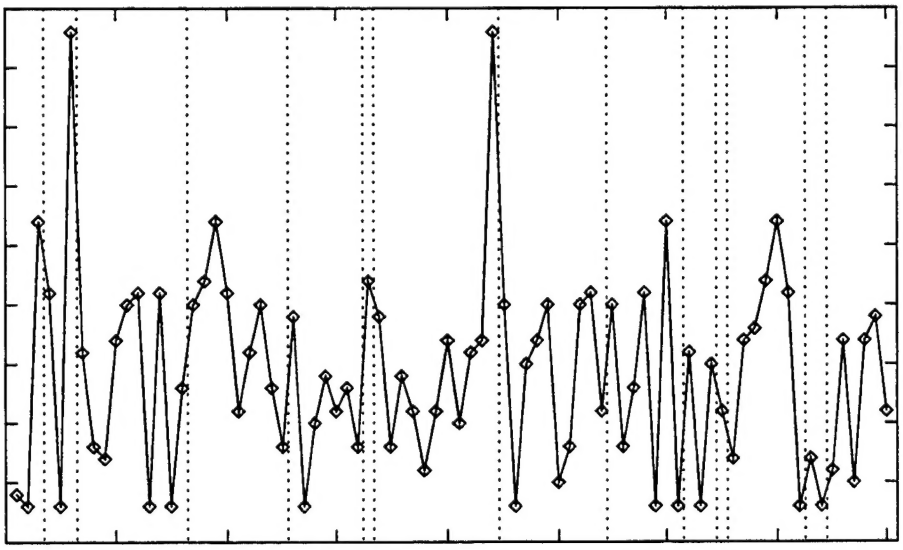


Figure 2. Time series representation.

In Figure 2, the first 80 letters (emboldened in Figure 1) of the Serbian text are displayed. The vertical dashed lines mark the location of interword spaces, which have been removed, along with most punctuation, to facilitate our methodology. The x-axis indexes the order of occurrence of the characters in the text, and the corresponding codeset values are plotted along the y-axis. If we allow a situation in which the characters are processed sequentially, then we can assign to each character an associated time epoch, and Figure 2 can be considered as a time series representation of the first 80 letters. The scale of measurement for the y-axis is nominal; an alternative codeset, if appropriate, would lead to a different graphic representation with no attendant loss or gain of information.

In attempting to generate a corpus, we would like a core of authentic documents to serve as a basis from which to generate additional pseudodocuments. An analogous situation, arising in the analysis of time series data collected as part of a clinical study, has been described and addressed using the bootstrap (Efron 1993, Liu 1992).

3. Bootstrap Application

In this section, we present an abridged description of the bootstrap procedure, modified for application to the textual model.* Notice the time series has an inherent structure: the time series represents a block of text—not a random sequence. Moreover, the words themselves are subject to lexical constraints; hence, the patterns they assume in the codeset representation have meaning. These word patterns are, however, interrupted with great frequency; the interword spaces play the role of interventions in time series modeling. As a consequence, the time series has local structure contributed by the word patterns but little in the way of global structure due to the high frequency of interventions. Any attempt to model these data, statistical or otherwise, must maintain the fidelity of the overall structure.

Denoting the time series as a sequence of ordered pairs $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, we begin the bootstrap procedure by choosing a random location within the time series, say (x_r, y_r). Starting with (x_r, y_r), we copy the subsequence $(x_r, y_r), (x_{r+1}, y_{r+1}), \ldots, (x_{r'}, y_{r'})$ and write to an array. The length of the subsequence, $r' - r + 1$, is determined by sampling from the distribution of word-lengths found in the authentic document. A second random location, (x_s, y_s), is then determined, and a second subsequence, $(x_s, y_s), (x_{s+1}, y_{s+1}), \ldots, (x_{s'}, y_{s'})$, is copied and appended to the subsequence already in the array. Figure 3 illustrates a situation in which three subsequences have been chosen, two of them overlapping.† The overlap does not create a problem since the sampling procedure is done with replacement. This process continues until terminated by a stopping rule. At that point, a bootstrapped time series, the first 80 values of which are shown in Figure 4, has been produced. The shaded regions appearing in Figure 3 are aligned in Figure 4 in order of their occurrence. Inverting the codeset mapping, subject to inherent lexical modeling constraints, yields the bootstrap document shown in Figure 5.

*See the appendix for an unabridged account of the computational paradigm.

†This example is somewhat contrived, in that the three subsequences were chosen from the first 80 characters pictured in Figure 2. In practice, all subsequences are randomly chosen within the entire document.

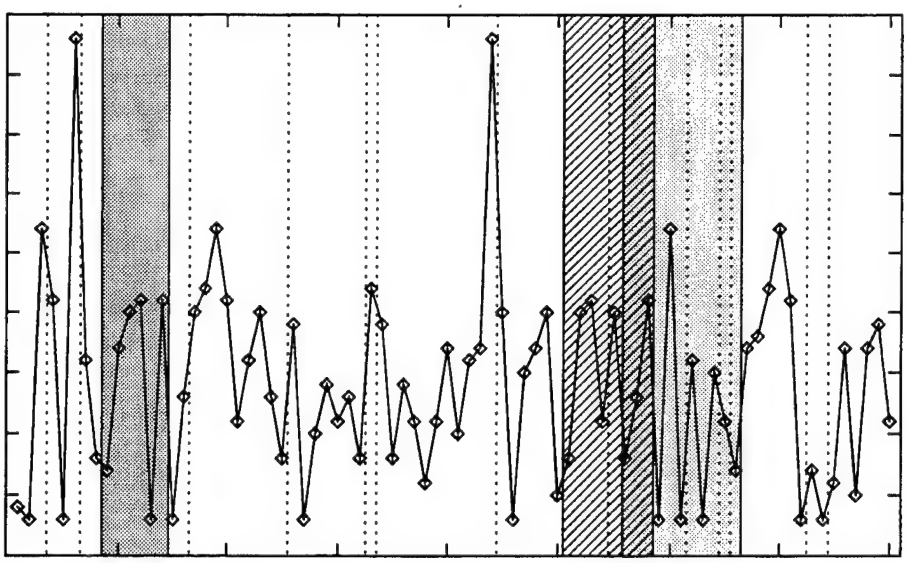


Figure 3. Intermediate results.

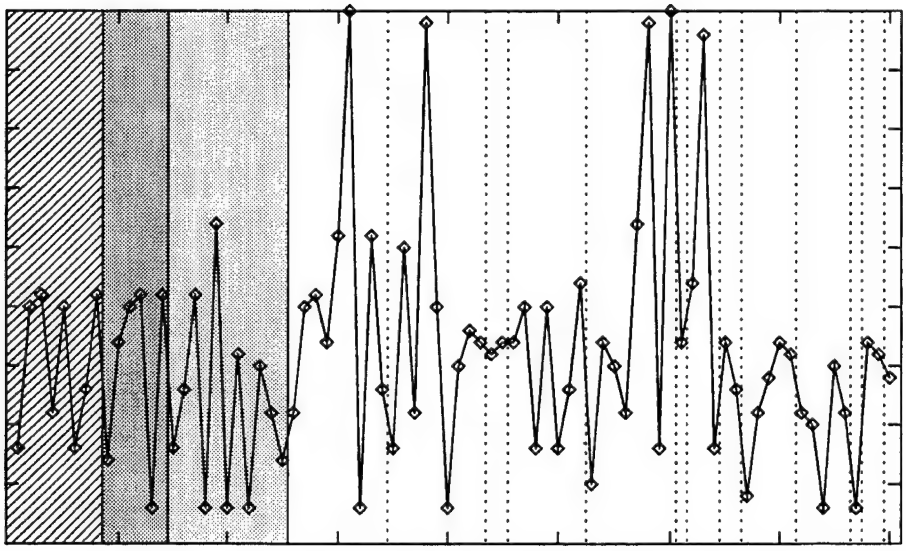


Figure 4. Bootstrapped time series.

Естисект достат екташанамидисточњачк ецилсампо но осесеку вомишљењ о ује ок билон изами А он ламибуд гизворасекта секташтв е ам фиран та нуде деизва ебр к оворим хљуд шансезамисио орад ра ииликаоради. Сточњачкам иг осеби н каориг ање, к гхришћан рак раткојису ри кивањек уалц елиги керелиг ичн наси монасл тициза в пустиво есекулар" еислкад ентичнифе ањеговеп, јумного цизамувек а п, су. "Гиознојсам "овепропастип, уо ража отипичнерел хљудизакључу ил енине геза јуипом аштвапоштој а гдр атисфа" игијс амосве "освести рење ће ог" звраткојис Тизамищек" и ега онарскоде. "Емдасепр тосемож" ералр нтел амо, говепро и Њемубисем ње ци уј с" а м ерал авањебри ек х "негоинај разв римооне ет ансезам анего, амаенгенер че с јскеф очњачки. Кес уопш сатисфа вибиор воосекта се тиин жеприм Ст нскеразликеур стицизамув ачким екулари илон и вајут аж гов иразв "м с.

Figure 5. Bootstrapped text.

4. Empirical Results

The bootstrap procedure under which the document in Figure 5 was constructed* precludes its being "read" by an individual. Our intent, however, was to produce a document image (or character string) sufficient to assess the character recognition capability of an OCR product. If the OCR software has incorporated language-specific decision aids to support character segmentation, the bootstrap document will likely reduce the effectiveness of those procedures. Clearly, spell-checkers will not be of value. Lexical analyzers (e.g., hidden-Markov models) will likely be degraded, but not rendered ineffectual, since substantial local structure has been retained under the moving-blocks procedure.

4.1 N-gram Profiles

There is a widely accepted statistical approach to automated language identification that does not rely on identifying words of a text (Cavnar 1994). This approach is based on the distribution of textual n-grams.† While we are not interested here in language identification, we are keenly interested in producing documents that remain indistinguishable from the actual language under these identification schemes.

*A modified moving-blocks bootstrap.

†The n-grams of a text are all the character sequences of length n contained in that text. For example, *special forces* contains 14 unigrams (s,p,e,...), 13 bigrams (sp,pe,ec,...), 12 trigrams (spe,pec,eci,...), and so on.

Toward that end, we have compared n-gram profiles of an original document against its bootstrap progeny. A typical result from such a comparison, in which the bigrams of five bootstrap replicates (labeled **boot1**,...,**boot5**) were individually compared with the bigrams of the original document, is shown in Figure 6. Bigrams whose frequency differed by less than 0.005 in absolute value from the original document for all five bootstrap replicates, $|f_{boot(i)} - f_{orig}| < .005, i = 1, \ldots, 5$, were not plotted. In this example, 7.6% of inner word bigram frequencies were determined to differ by more than this amount. Those instances are plotted in the left panel of Figure 6, where it can be seen that, for a given bigram, the inequality was often violated by only a single bootstrap replicate, and the difference was seldom in excess of 0.007.

An artifact of the moving-blocks bootstrap was the creation of bigrams that did not appear in the original document. These typically arose at the "edges" of bootstrap words, involving a bigram of the form (space, character) or (character, space).* Those occasions in which the inequality was violated for these spurious bigrams are pictured in the right panel of Figure 6. The annexing of data whose spatial dependencies across subregion boundaries do not reflect those in the original data set is at the core of this problem and has received research attention from several investigators (Hall 1985, Hall 1988, Kunsch 1989). The rejection rate for inner word and interword bigrams combined was 14%. This value is influenced, in addition to the threshold level, by document size since frequencies, $f_{(\cdot)}$, and document size are inversely proportional.

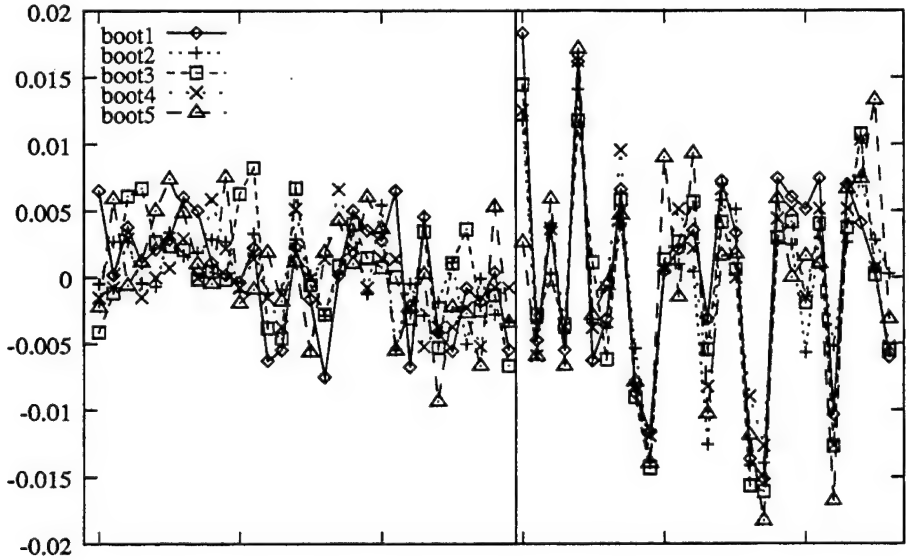


Figure 6. Frequency differences.

4.2 Chernoff Faces

The disparity among bigram frequencies shown in Figure 6 is only one of several physical and syntactic measures useful for comparison of documents. A number of such measures,

*Let ␣ represent an interword space. The edge bigrams of an arbitrary word **wxyz** are then ␣**w** and **z**␣.

corresponding to an authentic Serbian document and five bootstrap replicates, are displayed in Table 1.

Table 1. Comparison measures

doc	chars	words	lines	unigm	inrbigm	edgbigm	siginr	sigedg	sigbi
doc3	2865	395	45	35	254	46	0	0	0
boot3a	2839	400	44	34	280	57	2	16	18
boot3b	3294	453	50	35	283	55	2	13	15
boot3c	2633	370	43	35	282	55	5	14	19
boot3d	3543	488	54	35	293	59	3	14	17
boot3e	2327	338	37	34	257	56	5	14	19

Legend: *doc* = document identification. chars = number of characters. words = words. lines = lines of text. unigm = distinct unigrams. inrbigm = distinct inner bigrams. edgbigm = distinct edge bigrams. siginr = significant inner bigrams. sigedg = significant edge bigrams. sigbi = siginr + sigedg.

In column 1, doc3 identifies the original document; boot3a–e are the bootstrap replicates. Columns 2–4 are physical measurements, strongly correlated measures of document size. Columns 5–10 are syntactic measurements, and in particular, columns 8–9 are summaries of the content of Figure 6.

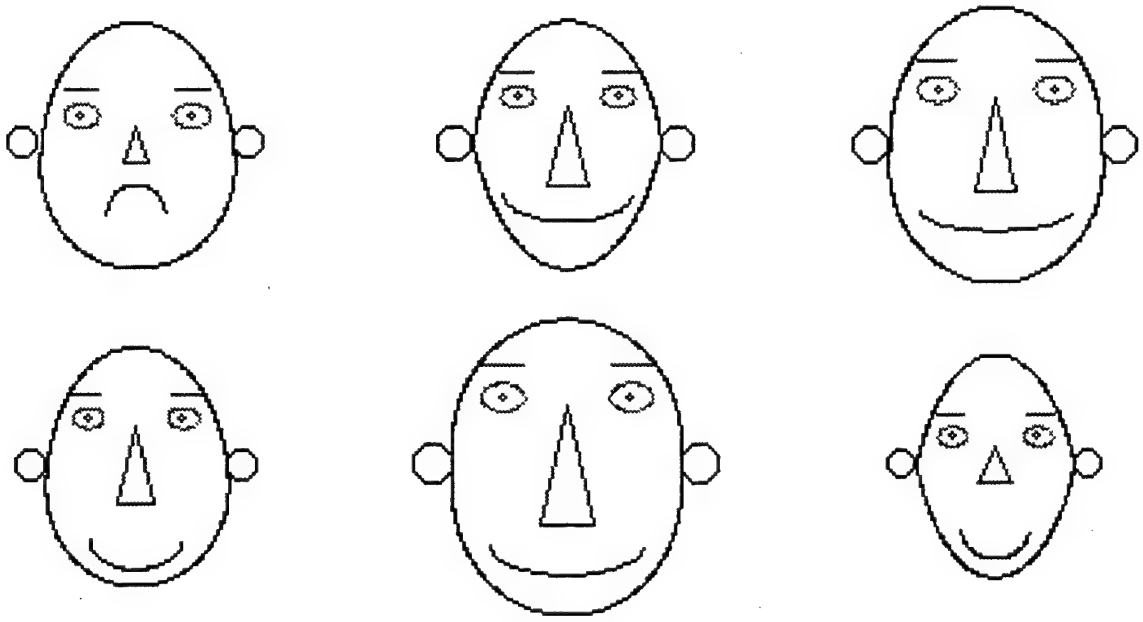
Visual display of multidimensional data—each document in Table 1 has associated with it nine values—is an active topic of research. One of the more innovative approaches involves the use of representative icons. Shown in Figure 7 for the data in Table 1 is one of the most widely accepted portrayals, called Chernoff faces.*

We knew a priori that the bootstrap process would not produce documents whose Chernoff-face icons mirrored the face representing the original document. The number of significant inner and edge bigrams will always be zero for the original document by virtue of construction. Those attributes we associated with the mouth, and the difference between original and bootstrap is striking. The number of lines, words, and characters will differ since the physical constraints we impose are deliberately relaxed to allow for randomness. These physical attributes are associated with the head configuration, where, in general, the area of the head conveys document size. The size of the nose represents the total number of inner and edge bigrams, where the minimum again corresponds to the original document.

4.3 Comparison of Character Accuracy

Five Serbian documents of comparable size were selected as the basis of a more intensive investigation. Groundtruth files were created for each of the documents through keyboard entry and post-verification. Three inquiries were then undertaken.

*Named after their originator H. Chernoff (Chernoff 1973).



Legend: face/w = chars. halfface/h = words. upface/ecc = lines. loface/ecc = unigm. nose/l = inrbigm. nose/w = edgbigm. mouth/cent = siginr. mouth/l = sigedg. mouth/curv = sigbi.

Figure 7. Chernoff faces.

First, the Serbian documents were scanned and submitted to the OCR software for segmentation; the groundtruth and OCR output files were compared for agreement using specialized scoring software (DoD 1997); as part of the comparison, the character accuracy for each of the five documents was determined. The percentage character accuracies, labeled **original**, are plotted in Figure 8.

The groundtruth files were then printed. The printer output was scanned, processed by the OCR module, and the results compared, via the DOD software, against the groundtruth files. Those percentage agreements, labeled **ground**, are again shown in Figure 8.

Finally, for each of the 5 Serbian documents, 5 bootstrap replicates were generated (25 bootstrap documents in all). At this juncture, manual intervention was required to reconcile, as much as possible, any differences in font style or point size that might exist between original and bootstrap documents and lead to an additional source of error. The bootstrap files were printed, and the hardcopy scanned and OCR'd. The bootstrap files and OCR output were compared, and the average percentage agreement, labeled **boot**, was plotted in Figure 8 along with the individual values making up the averages.

All five bootstrap replicates based on the second document, although in close percentage agreement, had accuracy assignments below that of the original document. While an analysis of the OCR accuracy in every instance is beyond the scope of this report, a brief explanation for this occurrence is in order. One Serbian letter was never correctly identified in these five

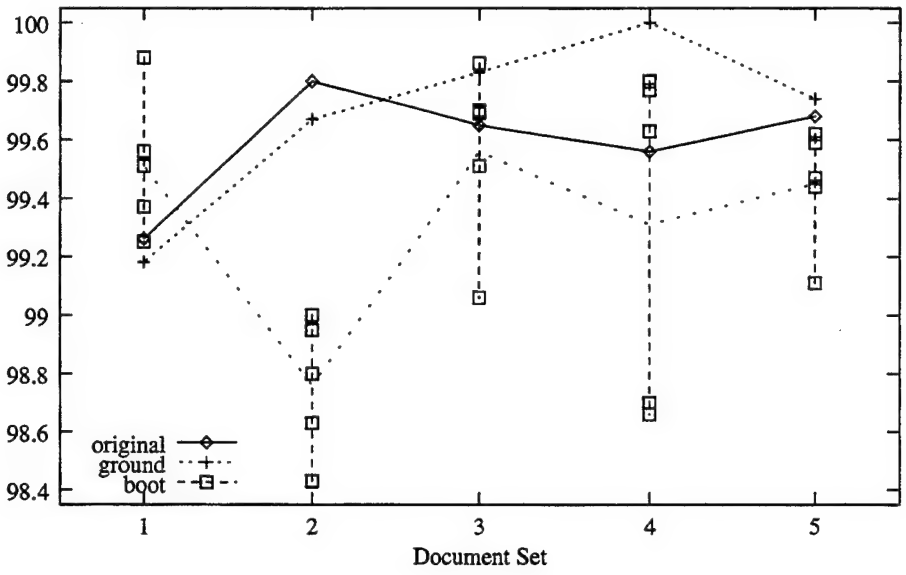


Figure 8. Character accuracy.

bootstraps, although it was always recognized in the original document.* If that one letter is ignored, the bootstrap document accuracies increase to a range of 99.24–99.70%, just below the original document's adjusted accuracy of 99.80%. The problem, in this instance, appears to involve the OCR module rather than the bootstrap technique.

Notice the overall range of percentages plotted in Figure 8—[98.4, 100]. For most practical purposes—certainly for the purpose of our investigation—the bootstrap documents can serve as a surrogate for the authentic Serbian documents. That was what we wanted to establish, but, as we have just seen, information of value to the developer may also spontaneously occur.

5. Model Validation

We have detailed in section 3 and the appendix the mechanics of producing a bootstrap document. The empirical results provided in section 4, while insightful and persuasive, still stop short of advancing a general procedure for rigorous assessment of a bootstrap document's ability to perform as a surrogate manuscript. Such a procedure is the topic of this section.

Up to now, we have used words like pseudodocument, surrogate, and progeny to describe the role intended for a bootstrap document. An expression we have not used, but equally appropriate, is "simulated document." We want to introduce that expression, and that notion, at this juncture. If the bootstrap document is thought of as a simulated document, then the procedure responsible for its existence is a simulation procedure. In other words, the modified moving-blocks bootstrap procedure may be considered the central part of a stochastic simulation model.

*Curiously, the same letter had reasonable recognition accuracy in the other four sets of documents!

The discussion to follow will be facilitated by the introduction of some additional notation and terminology.

Let $\mathbf{x} = (x_1, \ldots, x_p)$ be a vector of inputs parameterizing a stochastic simulation model. The inputs may be values of a mathematical variable, measurements on a random variable, or a combination of the two. *For our application, number of paragraphs, number of sentences, number of double quotes, sentence lengths, word lengths,... are all input parameters.* Let y denote the output of a simulation model: $y \in A$ takes on values in a set A determined by the model structure. Let z be a measurement on a real-world process being simulated, whose attributes coincide with those of the input vector $\mathbf{x}$. *For our application, y is the percentage measure of agreement between a bootstrap document and its groundtruth; z is the percentage measure of agreement between the authentic document and its groundtruth.* In general, $y \neq z$, since both y and z are observations on a random variable—y because the model is stochastic, and z because the model specification is incomplete. *For example, point size, font family, physical attributes of the paper, are all uncontrolled in the model under discussion.* For a fixed $\mathbf{x}$, many values of z may be observed, since some but not all of the relevant variables and relationships are represented in $\mathbf{x}$. Since the purpose of a simulation model is to mimic a real-world process, in attempting to validate the simulation, a comparison of empirical data with the model output generated for the same conditions, as represented through the vector $\mathbf{x}$, is required.

Suppose that n paired observations $(y_1, z_1), \ldots, (y_n, z_n)$ are available for comparison, where each pair corresponds to a simulation run with a different input vector. *Here, $(y_1, \ldots, y_n)$ are percentage accuracies for single bootstrap replicates; $(z_1, \ldots, z_n)$ are percentage accuracies for the corresponding groundtruth documents.* Since each pair was generated under different conditions, preliminary pooling of the data is inappropriate. A procedure that examines each pair individually, and then allows for the combination of these comparisons into an overall assessment is required.

For m runs of the simulation model with a fixed input vector $\mathbf{x_i}$, a set of output values $y_{i1}, \ldots, y_{im}$ that can be compared with a corresponding empirical value z_i is produced. Recall that $\mathbf{x}$ does not contain all of the relevant input variables. This means that z, for a specific value of $\mathbf{x}$, behaves as a random variable conditioned on $\mathbf{x}$. Likewise, y is a random variable conditioned on $\mathbf{x}$ by model construction. To validate a simulation model, a viable approach would be to establish that $\mathrm{F}(y \mid \mathbf{x})$, the conditional distribution of y, coincides with $\mathrm{G}(z \mid \mathbf{x})$, the conditional distribution of z, for $-\infty < y, z < \infty$, and $\mathbf{x} \in \mathbf{\Omega}$, a set of relevant inputs.

For m runs of the simulation model for each of n different input vectors $\mathbf{x_1}, \ldots, \mathbf{x_n}$, the resultant data configuration $(y_{11}, \ldots, y_{1m}; z_1), \ldots, (y_{n1}, \ldots, y_{nm}; z_n)$ may be treated as n multivariate observations, where the y_{ij} for fixed i are independent and identically distributed. If the components of the vector $(y_{i1}, \ldots, y_{im}; z_i)$ are ranked for each i, and, if the simulation model is valid, the rank assigned to z_i should be equally likely among the possible ranks $1, \ldots, m+1$. This notion finds implementation in the Mann-Whitney test, a nonparametric two-sample test for location.

Several independent Mann-Whitney tests can be combined through a statistical procedure known as a permutation test. The essence of a permutation test in the present application is as follows: Let R_i denote the rank of z_i in the i^{th} observation $(y_{i1}, \ldots, y_{im}; z_i)$ after the

components have been ordered from smallest to largest; R_i is an integer between 1 and $m+1$ inclusively. A test statistic T is defined as the sum of the R_is over all n observations; $T = \sum_{i=1}^{n} R_i$. Values of T that are determined to be too small or too large lead to rejection of the null hypothesis. The null hypothesis is that $\mathrm{F}(y \,|\, \mathbf{x}) = \mathrm{G}(z \,|\, \mathbf{x})$, for all $-\infty < y, z < \infty$, and $\mathbf{x} \in \mathbf{\Omega}$, which can be stated in words as "the simulation model is valid," or, *the bootstrap manuscript is indistinguishable from an authentic document in terms of OCR accuracy measurements.*

What remains is to quantify the expressions "too small" and "too large." To do this, we need to know what values the test statistic T might assume and with what frequency (probability) under the null hypothesis. This is most easily explained with a numerical example. The data described in section 4.3 and shown in Figure 8 are, after transforming to ranks, in the exact format required.

We will continue the discussion focusing on these data. Clearly, T can take on all integer values between 5 and 30, inclusively. Associating a frequency of occurrence with each value of T is a more daunting exercise. An exact solution requires the systematic enumeration of every possible permutation of ranks within the five vectors of dimension six: $(y_{i1}, \ldots, y_{i5}; z_i), i = 1, \ldots, 5$, and the evaluation of the corresponding statistic $T = \sum_{i=1}^{n} R_i$. That amounts to $(6!)^5 = 1.934917632 \times 10^{14}$ values in total.

Numbers of such magnitude may be excessive and impractical. A much smaller random sample taken from the set of all possible permutations may be adequate to construct a reference distribution for T (Edgington 1987). This was the case here. The resulting distribution of T, based on a random sample of 10^5 permutations, appears in Figure 9.*

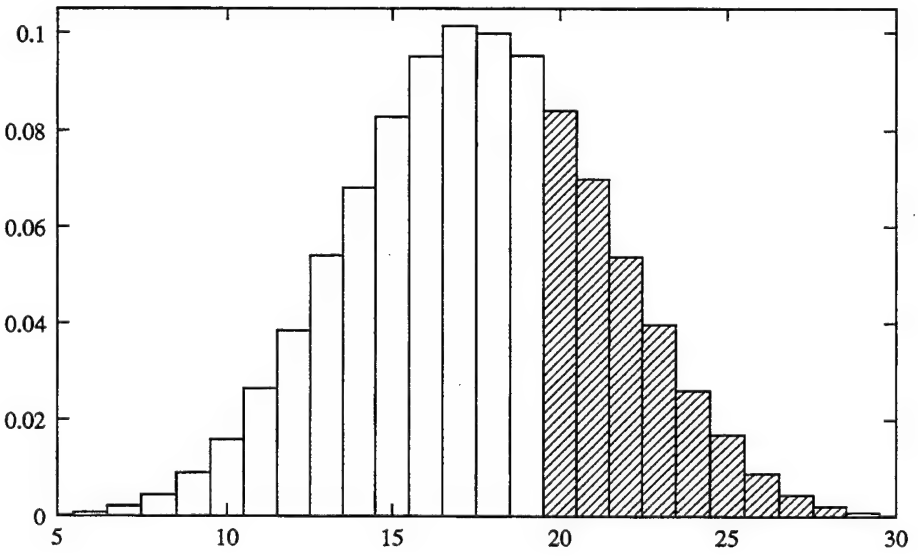


Figure 9. Reference distribution for T.

*A normal approximation to the distribution of T is sometimes adequate, depending on the permutation sample size and the number of ranks to be assigned.

The experimentally determined value of T, T=20, is seen to lie well inward of the reference distribution. As a matter of fact, values of T as large as we observed, or larger, will occur 31% of the time when the null hypothesis is valid. The statistic T is not nearly large enough to cause concern that our claim of indistinguishability might be in error. In the language of statistical hypothesis testing, we have an observed significance level of 0.31, compared to values of 0.05 or 0.01 traditionally chosen as levels to trigger rejection.

6. Summary

A modified moving-blocks bootstrap was applied to the construction of pseudodocuments used for evaluation of an OCR module. The n-gram profiles of the resultant bootstrap documents appeared to be consistent with that of the source-language document in a limited empirical study. A more extensive comparison of bootstrap and source-language documents via the OCR module produced no discernible distinction between the two classes. The procedure governing bootstrap document generation was validated using a rigorous statistical procedure. These results strengthen the advocacy of a statistical approach to corpus generation and encourage the implementation of more rigorous paradigms into the field of natural language processing.

7. References

Cavnar, William, and John Trenkle. "N-gram-Based Text Categorization." *Symposium on Document Analysis and Information Retrieval*, pp. 161–175, 1994.

Chernoff, H. "The Use of Faces to Represent Points in k-Dimensional Space Graphically." *Journal of the American Statistical Association*, vol. 68, pp. 361–367, 1973.

Department of Defense. Document Scoring Software Version 5.0. Fort Meade, MD, 1997.

Edgington, Eugene S. "Randomization Tests, 2nd Ed." *Statistics: Textbooks and Monographs, vol. 77*, New York, NY: Marcel Dekker, 1987.

Efron, Bradley, and Robert J. Tibshirani. "An Introduction to the Bootstrap." *Monographs on Statistics and Applied Probability, no. 57*, New York, NY: Chapman & Hall, 1993.

Hall, P. "Resampling a Coverage Pattern." *Stochastic Processes and Their Applications*, vol. 20, pp. 231–246, 1985.

Hall, P. "On Confidence Intervals for Spatial Parameters Estimated From Nonreplicated Data." *Biometrics*, vol. 44, pp. 271–277, 1988.

Kunsch, H.R. "The Jackknife and the Bootstrap for General Stationary Observations." *Annals of Statistics*, vol. 17, pp. 1217–1241, 1989.

Liu, R.Y., and K. Singh. "Moving Blocks Jackknife and Bootstrap Capture Weak Dependence." *Exploring the Limits of Bootstrap*, New York, NY: John Wiley & Sons, edited by LePage and Billard, 1992.

Reeder, Flo, and Jerry Geisler. "Multi-Byte Issues in Encoding/Language Identification." *Proceedings of Workshop on Embedded MT Systems: Design, Construction, and Evaluation of Systems with an MT Component*, held in conjunction with AMTA '98, pp. 49–58, Langhorne, PA, 1998.

INTENTIONALLY LEFT BLANK.

Appendix:

Program Methodology

INTENTIONALLY LEFT BLANK.

A.1 Goals

The goal of this program is to produce text, using the bootstrap method, that is both visually and syntactically comparable to an original document. While very little is language-specific, each language has its own syntax. The following explanation uses an English document and syntax.*

To begin, here are the assumptions and limitations we put on the text.

1. Sentences start with a capital letter and end with a period, exclamation mark, or question mark. When the sample file is processed, these three characters denote the end of a sentence.
2. One or more sentences make up a paragraph. Paragraphs are separated by a blank line.
3. Internal punctuation may appear at the end of a word, while a single quote (apostrophe) may appear anywhere within a word. We currently check for commas, semicolons, colons, and single quotes.
4. Proper nouns may appear anywhere. There are no acronyms.
5. Words may be hyphenated at the ends of lines. Internal hyphens (dashes) are ignored.
6. Numbers, other punctuation, and special symbols are ignored.

A.2 Preprocessing

We distinguish between the structure and content of a document. In order to exercise control over the appearance of a bootstrap document, certain parameter values are extracted from the original document and used as the basis for the formulation of global constraints. Consider the excerpt of text in Figure A-1.

The number of double quotes, the number of lines that end with a hyphen, and the number of paragraphs are all recorded. The maximum line length is also determined to properly format the output. To make it easier to compute the sentence lengths, we replace everything—except letters, sentence ends, and single quotes—with a space, collapsing multiple spaces into a single space. We record the number of words in each sentence and the total number of capitalized words. The set of characters that end each sentence is also retained. (The resultant text is shown in Figure A-2.)

We now have empirical values describing the structure of the document. In this example, there is only a single paragraph. The set of sentence lengths, measured in words per sentence, is {28, 34, 35, 27, 29, 28}, and the sentences end with five periods and a question mark,

*The main body of this report employed a variant of this program that was modified to manipulate Serbian (Cyrillic) text.

```
After about an hour of this amusement, in the latter part of
which Job didn't participate, the mutes by signs indicated that
Billali was waiting for an audience. Accordingly he was told to
"crawl up," which he did as awkwardly as usual, and announced
that the dance was ready to begin if She and the white strangers
would be pleased to attend. Shortly afterwards we all rose, and
Ayesha having thrown a dark cloak (the same, by the way, that she
had worn when I saw her cursing by the fire) over her white
wrappings, we started. The dance was to be held in the open air,
on the smooth rocky plateau in front of the great cave, and
thither we made our way. About fifteen paces from the mouth of
the cave we found three chairs placed, and here we sat and
waited, for as yet no dancers were to be seen? The night was
almost, but not quite, dark, the moon not having risen as yet,
which made us wonder how we should be able to see the dancing.
```

Figure A-1. Original text.

{. . . . ? .}. The capitalized word total less the number of sentences, $11 - 6 = 5$, provides an approximation to the number of proper nouns. The text contains two double quotes and no hyphenated words. The maximum line length, or text width, is 66 characters.

Returning to the original text (Figure A-1), we convert all letters to lower case. Everything except letters and internal punctuation is replaced with a blank; multiple blanks are again collapsed into single blanks to produce the contents of Figure A-3. The last piece of empirical information we need is the set of word lengths. For our purposes, a word is any sequence of letters, or letters and internal punctuation. The example has words of length {5, 5, 2, 4, ..., 3, 3, 7}.

All of the words, including internal punctuation, are concatenated into a single string as shown in Figure A-4. The text is shown as a block to emphasize the fact that it is a single, very long line. This sequence of characters, or its codeset representation, is the time series that we are going to bootstrap to produce a new document.

A.3 Bootstrap Mechanics

To determine the number of sentences that will comprise the bootstrap document, we sample from a distribution of the form shown in the left side of Figure A-5, whose median is set equal to the number of sentences in the original document, and whose range of values covers the potential choices for this attribute. Notice that the most likely values are 5, 6, and 7; we are going to generate 5 new sentences.

Next we decide how many words should appear in each sentence. To determine this, a value is drawn with replacement from the set of sentence lengths previously recorded. The

After about an hour of this amusement in the latter part of which Job didn't participate the mutes by signs indicated that Billali was waiting for an audience. Accordingly he was told to crawl up which he did as awkwardly as usual and announced that the dance was ready to begin if She and the white strangers would be pleased to attend. Shortly afterwards we all rose and Ayesha having thrown a dark cloak the same by the way that she had worn when I saw her cursing by the fire over her white wrappings we started. The dance was to be held in the open air on the smooth rocky plateau in front of the great cave and thither we made our way. About fifteen paces from the mouth of the cave we found three chairs placed and here we sat and waited for as yet no dancers were to be seen? The night was almost but not quite dark the moon not having risen as yet which made us wonder how we should be able to see the dancing.

Figure A-2. First pass.

after about an hour of this amusement, in the latter part of which job didn't participate, the mutes by signs indicated that billali was waiting for an audience accordingly he was told to crawl up, which he did as awkwardly as usual, and announced that the dance was ready to begin if she and the white strangers would be pleased to attend shortly afterwards we all rose, and ayesha having thrown a dark cloak the same, by the way, that she had worn when i saw her cursing by the fire over her white wrappings, we started the dance was to be held in the open air, on the smooth rocky plateau in front of the great cave, and thither we made our way about fifteen paces from the mouth of the cave we found three chairs placed, and here we sat and waited, for as yet no dancers were to be seen the night was almost, but not quite, dark, the moon not having risen as yet, which made us wonder how we should be able to see the dancing

Figure A-3. Second pass.

afteraboutanhourofthisamusement,inthelatterpartofwhichjobdidn'tpa
rticipate,themutesbysignsindicatedthatbillaliwaswaitingforanaudie
nceaccordinglyhewastoldtocrawlup,whichhedidasawkwardlyasusual,and
announcedthatthedancewasreadytobeginifsheandthewhitestrangerswoul
dbepleasedtoattendshortlyafterwardsweallrose,andayeshahavingthrow
nadarkcloakthesame,bytheway,thatshehadwornwhenisawhercursingbythe
fireoverherwhitewrappings,westartedthedancewastobeheldintheopenai
r,onthesmoothrockyplateauinfrontofthegreatcave,andthitherwemadeou
rwayaboutfifteenpacesfromthemouthofthecavewefoundthreechairsplace
d,andherewesatandwaited,forasyetnodancersweretobeseenthenightwasa
lmost,butnotquite,dark,themoonnothavingrisenasyet,whichmadeuswond
erhowweshouldbeabletoseethedancing

Figure A-4. Concatenated text.

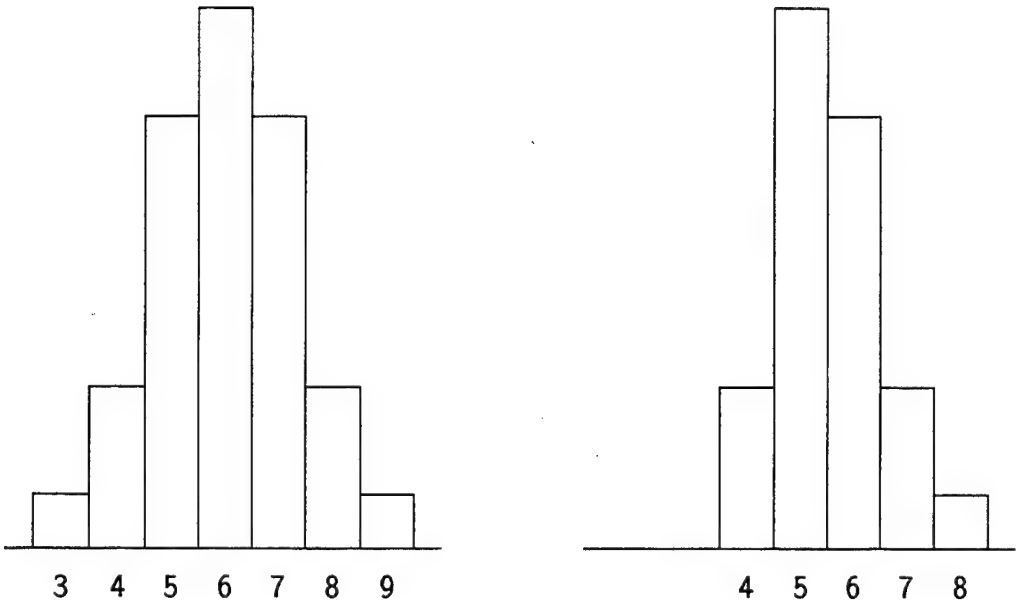


Figure A-5. Histograms modeling number of sentences (left) and number of proper nouns (right).

example shown here uses the values 28, 27, 29, 28, and 28. The lengths of each word within a sentence are chosen in a similar manner. Values are drawn with replacement from the set of word lengths extracted from the original document. We are going to use the values 5, 9, 3, 5, ..., 3, 4, 3.

Having determined the bulk of the document's structure, we are now ready to sample the time series. Turning to the character string (Figure A-4), we determine a random location within the string and a random word length. The sequence of characters commencing at the random location (660) and continuing through the random length (5) will comprise the first word (otqui) of the first sentence.* Should the choice of random location and word length cause us to extend beyond the end of the string, we choose another random location. The first four snippets obtained in this manner are shown in Figure A-6.

```
afteraboutanhourofthisamusement,inthelatterpartofwhichjobdidn'tpa
rticipate,themutesbysignsindicatedthatbill aliwa[4]swaitingforanaudie
nceaccordinglyhewastoldtocrawlup,whichhedidasawkwardlyasusual,and
announcedthatthedancewasreadytobeginifsheandthewhitestrangerswoul
dbepleasedtoattendshortlyafterwardsweallrose,andayes hahavingt[2]hrow
nadarkcloakthesame,bytheway,thatshehadwornwhenisawhercursingbythe
fireoverherwhitewrappings ,we[3]startedthedancewastobeheldintheopenai
r,onthesmoothrockyplateauinfrontofthegreatcave,andthitherwemadeou
rwayaboutfifteenpacesfromthemouthofthecavewefoundthreechairsplace
d,andherewesatandwaited,forasyetnodancersweretobeseenthenightwasa
lmost,butn otqui[1]te,dark,themoonnothavingrisenasyet,whichmadeuswond
erhowweshouldbeabletoseethedancing
```

Figure A-6. Snippets.

If a snippet contains punctuation, it may be manipulated slightly to make it conform to proper English syntax. If the first or last character in a snippet is a single quote, it is deleted. If the snippet contains internal punctuation, we move the first punctuation character to the end of the word and delete any others that may appear. For example, the snippet "ite,dark;th" would be changed to "itedarkth,", while the third snippet in Figure A-6 has the comma moved to the end, converting the text ",we" to "we,".

Should the snippet contain no letters, it is discarded and a new one extracted.

After all the words that make up a sentence have been extracted and, if necessary, modified, we capitalize the first word. An end-of-sentence character, randomly chosen from the set collected during preprocessing, is appended to the last word. This sequence of steps is repeated until we have generated the desired number of bootstrap sentences. The result is shown in Figure A-7 with arbitrary line breaks added.

*We refer to these character sequences as "snippets." While informal, it does convey the notion of what the procedure is about. We are extracting regenerative sequences (snippets) of random length and concatenating them—after appropriate attention to interword spaces, punctuation, and capitalization—to form sentences in a bootstrap document.

```
Otqui hahavingt we, aliwa madeu cursingbyt da
bythefireo thed westartedth ncerswere ce forana rasy
upwh, outfi ed ow sata ted dar ockyplateau eto dan, esf
ldbeple yhe syetnod. An inthe, sbysignsi ge ck artofwh
nd eg wasto moo ire wes cew igns hav ar three hortlya
tfi ro cav thatbill he tha, rd aui edt. Avewe she indi
ala, tha cur nthesmo eop the fthe ienc grisen egrea
echairs estr avi yhe ngerswo gnsindicated wh estarte
astobeh in eac arti ipa heh asyet ir. Ance tq acco
eheld ewa rosea, of tobe dan hou ngthrowna ace ethe
chairs andh, asready dthitherwema ed, ased nifshea thof
kth, heha swo epl ofth kclo most. At ment uldbepl ofwh
th th beh aces dth ea, verh smoothr rkcl syet, lyaft
emo ofthecavewef dbeable tesbysigns es dinglyhew oat
eple fift ngf ady atca fro.
```

Figure A-7. Intermediate sentences.

A.4 Postprocessing

The bootstrap text looks like a "real" document, but it needs further refinement. The first step is to capitalize some randomly chosen words to simulate proper nouns. The approximate number of proper nouns has already been determined. Since some sentences may have started with a proper noun, our count may be low. To compensate for this, we sample from a positively skewed distribution for proper noun total. The histogram used to model the number of proper nouns is shown on the right side of Figure A-5. In this example, five words were capitalized.

A random number of double quotes is inserted into the text. The number of quotes to add is determined by sampling from a distribution similar to that used for the number of sentences. The only difference is that the median value becomes the number of double quotes in the original document. In the example, we added three double quotes. For each quote, a word is randomly selected, then the double quote is randomly prepended or appended to the word. We do not require the double quotes to appear in matching pairs.

To break the text into paragraphs, we compute the probability that a given sentence terminates a paragraph, which is the number of paragraphs divided by the number of sentences. In this example, that is $\frac{1}{6}$ or 17%. For each sentence, we randomly generate a number from 0 to 99. If this number is less than 17, we start a new paragraph.

The final step is to format the text into the proper width. The words making up each sentence are combined into lines of text whose width cannot exceed the maximum line length of the original text. A blank is inserted between each pair of words, and two blanks are inserted between sentences. If any lines were hyphenated in the original text, a similar number of

hyphens is appended to randomly chosen lines. Figure A-8 shows the resultant text. Absent a literacy in English, the authentic and bootstrap documents are indistinguishable.

```
Otqui hahavingt we, aliwa Madeu cursingbyt da bythefireo thed
westartedth ncerswere ce forana rasy upwh, outfi ed ow sata" ted
dar ockyplateau eto dan, esf ldbeple yhe syetnod.  An inthe,
sbysignsi Ge ck artofwh nd eg wasto moo ire wes cew igns hav ar
three hortlya tfi ro cav thatbill he tha, rd aui edt.  Avewe she
indi ala, tha cur nthesmo "eop the fthe ienc grisen egrea echairs
estr avi yhe ngerswo gnsindicated wh estarte astobeh in eac arti
ipa heh asyet ir.  Ance tq acco eheld ewa rosea, of tobe dan hou
ngthrowna ace Ethe chairs andh, asready dthitherwema ed, ased
nifshea thof Kth, heha swo epl ofth kclo most.  At ment uldbepl
Ofwh th th beh aces dth ea, verh smoothr rkcl syet, lyaft "emo
ofthecavewef dbeable tesbysigns es dinglyhew oat eple fift ngf
ady atca fro.
```

Figure A-8. Bootstrapped text.

A.5 Further Enhancements

Section A.3 details how we bootstrap English text or, more properly, *Latinic* text. The same techniques may be used with other languages and other alphabets. The only real difficultly is converting between upper and lower case. This is trivial with the English-based program we used, but we had to explicitly list the character values in the Cyrillic version.*

We chose to ignore characters other than letters, certain internal punctuation, and end-of-sentence symbols. Our concern was to evaluate the accuracy of a Serbian (Cyrillic) OCR package, with emphasis on Cyrillic letters. It would not be difficult to add code to manipulate symbols that appear in pairs, such as parentheses, brackets, and braces.† In fact, the same technique could be used to insert double quotes in matching pairs within a single sentence.

Numbers are ignored in the current bootstrap program because it would be incorrect to extract a snippet that contained both letters and numbers. However, they could be treated as "number words" and processed in a manner similar to text words.‡ Acronyms were ignored because they would be interpreted as proper nouns. They could be counted, and a number of words could be converted to all upper case. Internal hyphens (compound words) could replace random blanks in the formatted text before it is printed.

*The program must be modified for each codeset.

†Our initial set of documents contained no brackets or braces, and only one document had parentheses.

‡Count the number of words and their lengths, then randomly generate a similar number by sampling the empirical lengths and randomly selecting digits. There is no need to use snippets with numbers.

As we have indicated, there are many possible refinements to the process. The added complexity must be weighed against the benefits gained. If the sample documents do not contain certain characters, there is no need to check for them. Syntactic accuracy is required only to the extent that it must conform to the language in the document. For example, a comma may appear only at the end of a word, not in the middle. The OCR software's accuracy is to be determined by having it process realistic documents. The program we have developed provides such documents.

NO. OF COPIES	ORGANIZATION
2	DEFENSE TECHNICAL INFORMATION CENTER DTIC DDA 8725 JOHN J KINGMAN RD STE 0944 FT BELVOIR VA 22060-6218
1	HQDA DAMO FDT 400 ARMY PENTAGON WASHINGTON DC 20310-0460
1	OSD OUSD(A&T)/ODDDR&E(R) R J TREW THE PENTAGON WASHINGTON DC 20301-7100
1	DPTY CG FOR RDA US ARMY MATERIEL CMD AMCRDA 5001 EISENHOWER AVE ALEXANDRIA VA 22333-0001
1	INST FOR ADVNCD TCHNLGY THE UNIV OF TEXAS AT AUSTIN PO BOX 202797 AUSTIN TX 78720-2797
1	DARPA B KASPAR 3701 N FAIRFAX DR ARLINGTON VA 22203-1714
1	NAVAL SURFACE WARFARE CTR CODE B07 J PENNELLA 17320 DAHLGREN RD BLDG 1470 RM 1101 DAHLGREN VA 22448-5100
1	US MILITARY ACADEMY MATH SCI CTR OF EXCELLENCE DEPT OF MATHEMATICAL SCI MADN MATH THAYER HALL WEST POINT NY 10996-1786

NO. OF COPIES	ORGANIZATION
1	DIRECTOR US ARMY RESEARCH LAB AMSRL D D R SMITH 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRL DD 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRL CS AS (RECORDS MGMT) 2800 POWDER MILL RD ADELPHI MD 20783-1145
3	DIRECTOR US ARMY RESEARCH LAB AMSRL CI LL 2800 POWDER MILL RD ADELPHI MD 20783-1145
	ABERDEEN PROVING GROUND
4	DIR USARL AMSRL CI LP (BLDG 305)

NO. OF
COPIES ORGANIZATION

2 DEPARMENT OF DEFENSE
G VAN DOREN
R522 STE 6514
FT MEADE MD 20755-6514

2 MITRE CORPORATION
F REEDER
M S W640
1820 DOLLY MADISON BLVD
MCLEAN VA 22102

2 UNIVERISTY OF MARYLAND
INST FOR ADVCED COMP STUDIES
T KANUNGO
4449 A V WILLIAMS BLDG
COLLEGE PARK MD 20740

ABERDEEN PROVING GROUND

44 DIR USARL
AMSRL CI
N RADHAKRISHNAN
J GANTT
AMSRL CI C
J GOWENS
AMSRL CI CD
B BODT
AMSRL CI CN
M HOLLAND (5 CPS)
C SCHLESIGER
C VOSS
AMSRL CI CT
A BRODEEN (15 CPS)
F BRUNDICK (15 CPS)
A CELMIŅŠ
AMSRL SL BD
W BAKER
AMSRL WM BC
D WEBB

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 2000	Final, Jun 98–Sep 99

4. TITLE AND SUBTITLE
A Statistical Approach to the Generation of a Database for Evaluating OCR Software

5. FUNDING NUMBERS
9FE320

6. AUTHOR(S)
Frederick S. Brundick, Ann E. M. Brodeen, and Malcolm S. Taylor

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: AMSRL-CI-CT
Aberdeen Proving Ground, MD 21005-5067

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-2265

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)

10.SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

In this report, we consider a statistical approach to augment a limited database of groundtruth documents for use in evaluating optical character recognition (OCR) software. We require groundtruth documents to assign a performance measure to the OCR component of the Forward Area Language Converter (FALCon) system. A modified moving-blocks bootstrap procedure is used to construct surrogate documents for this purpose which prove to serve effectively, and in some regards, indistinguishably from groundtruth. The proposed method is validated through a rigorous statistical procedure.

14. SUBJECT TERMS
bootstrap, time series, linguistics

15. NUMBER OF PAGES
28

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

INTENTIONALLY LEFT BLANK.

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. ARL Report Number/Author ARL-TR-2265 (Brundick) Date of Report July 2000

2. Date Report Received _____

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.) _____

4. Specifically, how is the report being used? (Information source, design data, procedure, source of ideas, etc.) _____

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided, or efficiencies achieved, etc? If so, please elaborate. _____

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.) _____

CURRENT ADDRESS

_____
Organization

_____
Name E-mail Name

_____
Street or P.O. Box No.

_____
City, State, Zip Code

7. If indicating a Change of Address or Address Correction, please provide the Current or Correct address above and the Old or Incorrect address below.

OLD ADDRESS

_____
Organization

_____
Name

_____
Street or P.O. Box No.

_____
City, State, Zip Code

(Remove this sheet, fold as indicated, tape closed, and mail.)
(DO NOT STAPLE)